CS580 ONLY Project # 4 Fall 2021

Group Project

Student Names	
Section	*CS 580*

For this project you will be researching transaction processing and concurrency control techniques (Ch. 21-22 in both 6th & 7th edition of textbook). This project consists of both a small research summary of the issues of this topic as well as completing problems related to this topic (below). This is a group project, but be sure all know how to do the problems below as understanding of them will be on the final exam.

SUBMIT:

One word document called P4-LastNamesOfGroupMembers.
In the document include names of students in the group and provide the following:

- A 5 page write up summary of the issues and challenges of transactions and concurrency; and the methods used to handle these challenges. This write up should include both what are the theoretical challenges, but also a description of how such a scenario could occur in a real application.
- The answers to the problems below:

STUDENT NAME(s):

1) Construct the serializability (or precedence) graph for the schedule specified bellow. Determine if the following schedule is (conflict) serializable. If it is, specify equivalent serial schedule(s).
r2(X); r3(X); w2(X); r1(X); w3(X)

2) Construct the serializability (or precedence) graph for the schedule specified bellow. Determine if the following schedule is (conflict) serializable. If it is, specify equivalent serial schedule(s).
r1(X); r2(X); w2(X); w1(X); r3(X)

3) Construct the serializability (or precedence) graph for the schedule specified bellow. Determine if the following schedule is (conflict) serializable. If it is, specify equivalent serial schedule(s).
r3(X); r2(X); w3(X); r1(X); w1(X), w2(X)

4) Construct the serializability (or precedence) graph for the schedule specified bellow. Determine if the following schedule is (conflict) serializable. If it is, specify equivalent serial schedule(s).
r3(X); r2(X); w3(X); w1(X)

5) Consider the three transactions T1, T2, and T3, and the schedule

specified bellow. Construct the serializability (precedence) graph. Determine if the schedule is (conflict) serializable. If it is, specify equivalent serial schedule(s).

T1: r1(X); r1(Z); w1(X);
T2: r2(Z); r2(Y); w2(Z); w2(Y);
T3: r3(X); r3(Y); w3(Y);

SCHEDULE: r1(X); r2(Z); r1(Z); r3(X); r3(Y); w1(X); w3(Y); r2(Y); w2(Z); w2(Y);